Access Barrier IOT Project

Corrado Parisi & Elias Schorr

Use-case

Wir nennen es Access Barrier. Wenn man die Küche während einer "forbidden time" betritt, lösen 2 Ranger aus und ein Buzzer ertönt, LED blinkt.

Wir stellen "entry" und "exit" anhand beider Ranger fest. Diese Infos senden wir bei jedem Trigger an z.Bsp ThingSpeak und können damit erfassen wann Personen allgemein in die Küche gegangen sind und wann "breaches" waren.

Der Alarm kann jeweils nur über einen Raspi in einem anderen Raum deaktiviert werden.

Interface documentation

Raspi interprocess communication

- `curl -X GET http://raspi2:5000/trigger_alarm`
- `curl -X GET http://raspi1:5000/kill_alarm`

Mqtt interface documentation

Host: mqtt.thingspeak.com

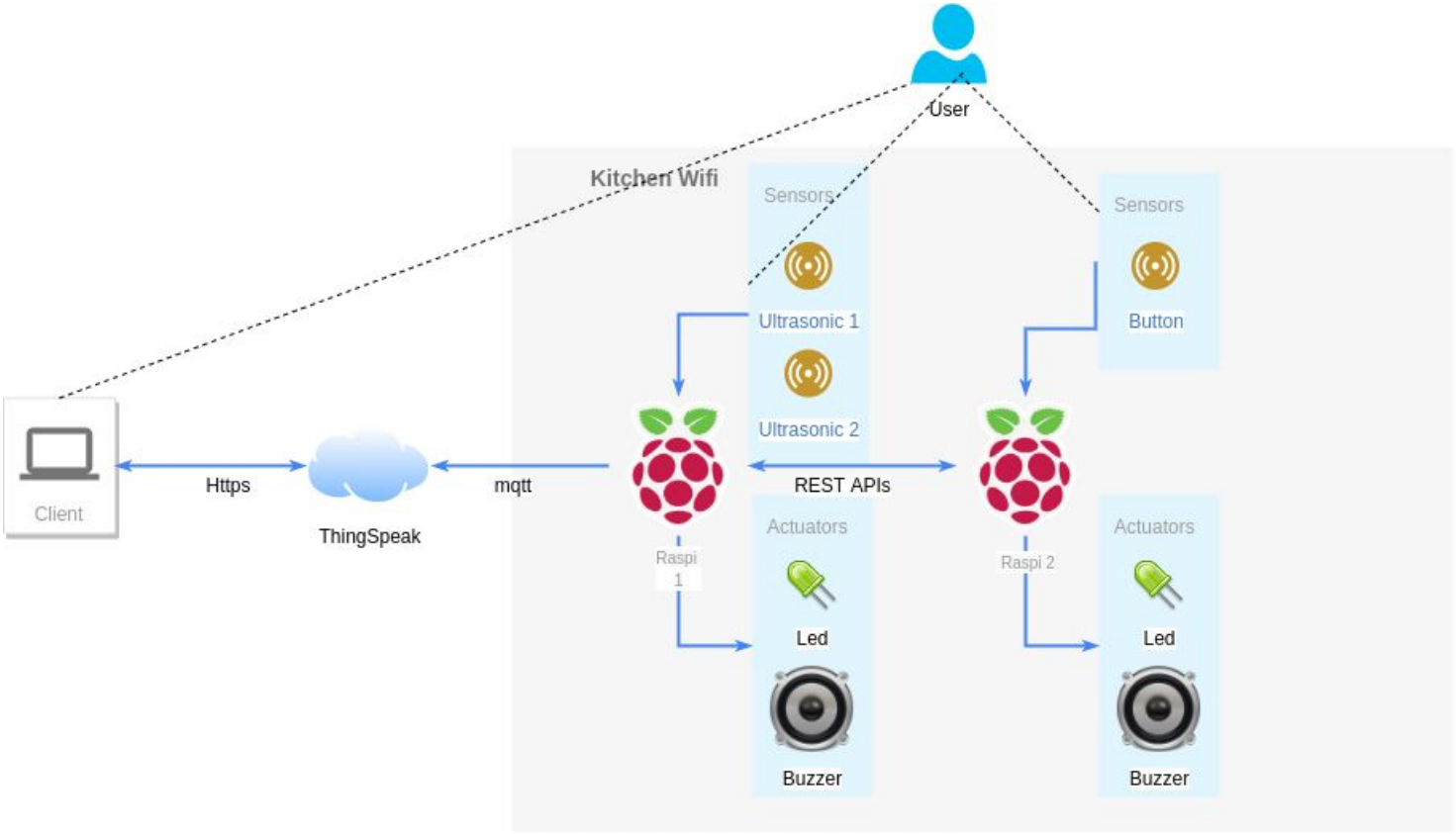
Port: 443

PUB -t 'channels/CHANNEL_ID/publish/WRITE_API_KEY' -m <’field1=1’ (room entry)

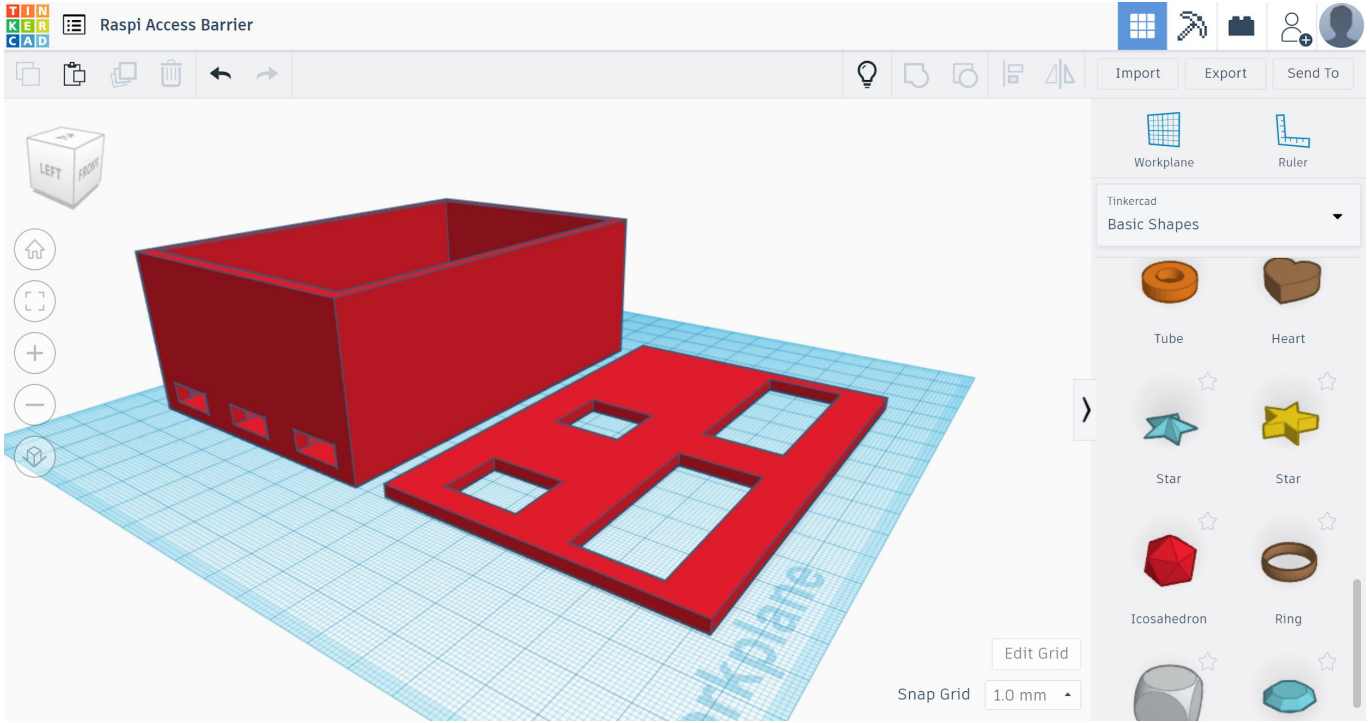
PUB -t 'channels/CHANNEL_ID/publish/WRITE_API_KEY' -m <’field2=1’ (room exit)

PUB -t 'channels/CHANNEL_ID/publish/WRITE_API_KEY' -m <’field3=1’ (alarm killed)

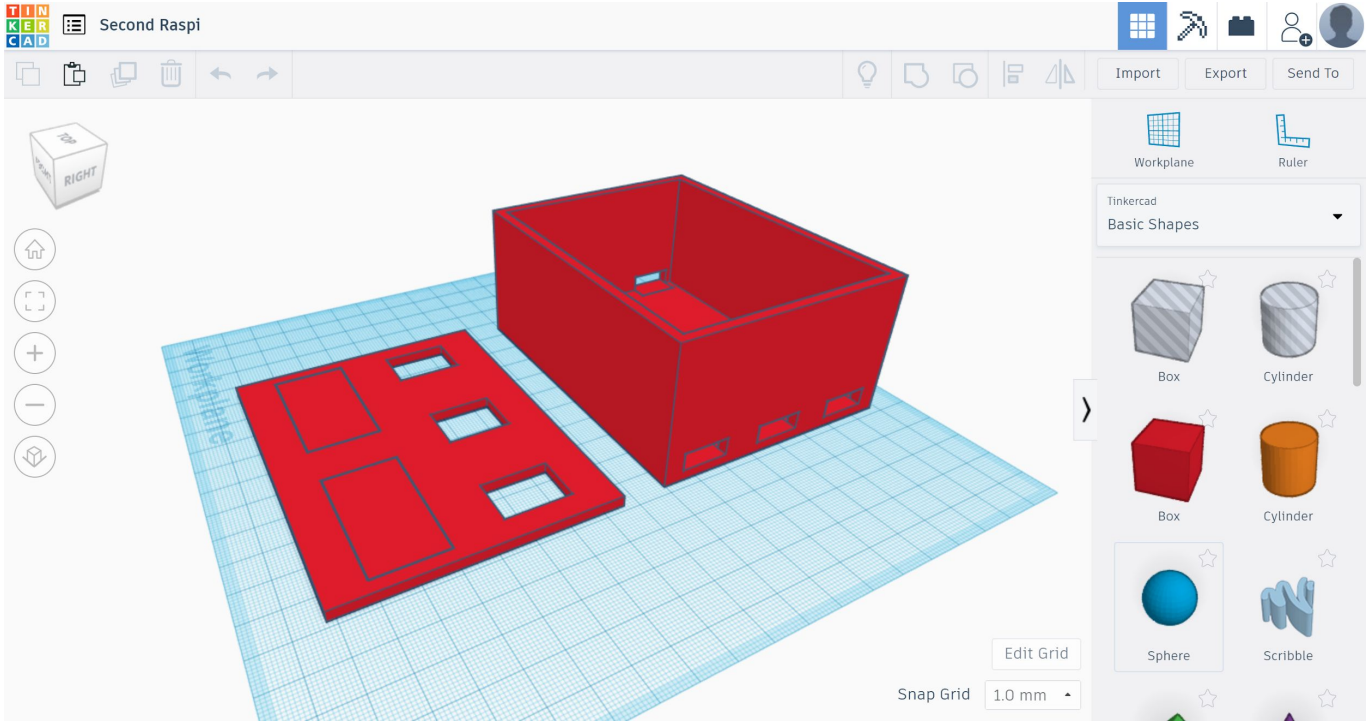
Reference model



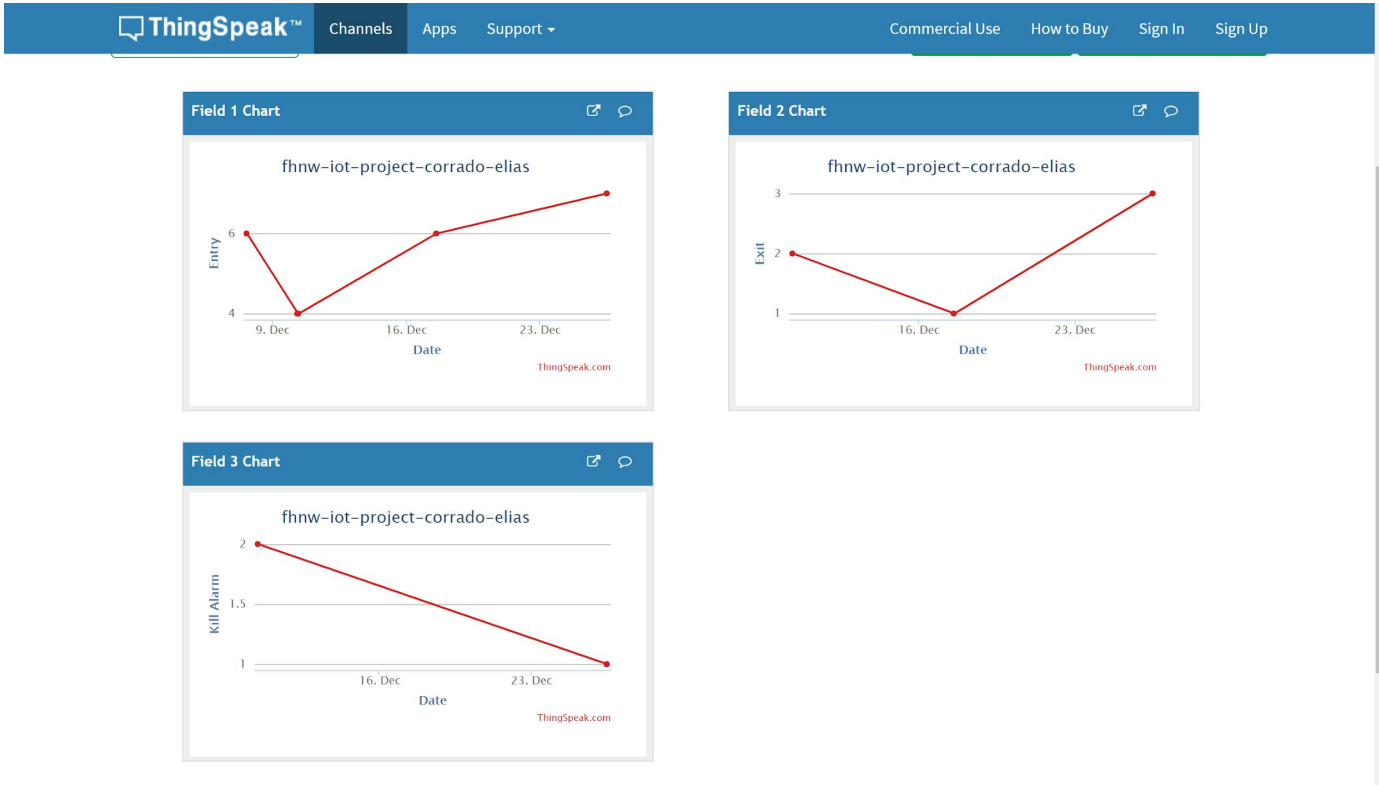
Raspi 1 (tinkercad)



Raspi 2 (tinkercad)




LIVE DEMO


ThingSpeak™
Channels
Apps
Support
Commercial Use
How to Buy
Sign In
Sign Up
Field 1 Chart
fhnw-iot-project-corrado-elias
Entry
6
4
9. Dec
16. Dec
23. Dec
Date
ThingSpeak.com
Field 2 Chart
fhnw-iot-project-corrado-elias
Exit
3
2
1
16. Dec
23. Dec
Date
ThingSpeak.com
Field 3 Chart
fhnw-iot-project-corrado-elias
Kill Alarm
2
1.5
1
16. Dec
23. Dec
Date
ThingSpeak.com

Issues

- At first we wanted to use nodejs(https://github.com/DexterInd/GrovePi/tree/master/Software/NodeJS) which required an outdated version of nodejs so we opted for python (https://github.com/DexterInd/GrovePi/tree/master/Software/Python).
- Windows has a problem with ext4 partitions, which corrupted a flashcard of a Raspberry Pi.